

# Flow-Based Monitoring and Analysis (FloMA)

Simon Leinen  
SWITCH  
<simon@switch.ch>

September 10, 1999

## Abstract:

This work package investigates the class of "flow-based" accounting methods for IP (Internet Protocol) traffic, and their applicability to the design and operation of large backbone networks such as TEN-155 or a National Research Network (NRN).

## Introduction

Traffic measurement has traditionally been an important component of network design, capacity planning, and operational monitoring. In the past, such measurements have generally been restricted to relatively simple metrics per interface or trunk, such as total input and output traffic rates. Thus, many interesting aspects of network traffic were relatively hard to measure. Examples of those are: the "application mix" of the traffic over a particular trunk line; distribution-valued parameters such as packet-size distributions; transport-oriented measures such as TCP connection throughput; classification of traffic on trunk lines according to customers etc.

In recent years, flow-based accounting mechanisms have been proposed that can be used to provide these types of information. One example is the IETF (Internet Engineering Task Force) RTFM (Real-Time Flow Measurement) architecture proposed by Nevil Brownlee of the University of Auckland. Another example is Cisco's *NetFlow* accounting.

As these systems reach maturity, and an abundance of software packages building upon them started to become available, this technology has become very interesting to backbone IP network operators recently. The goal of this work package is to evaluate available systems against the particular needs of backbone network operators.

In section [2](#), we will describe the test strategy adopted. The current (as of early September, 1999) status of the work is described in section [3](#). The accounting-based applications under investigation are outlined in section [4](#). This is followed by section [5](#) contained the results achieved so far. Planned future steps are enumerated in section [6](#), and section [7](#) concludes this paper with a short summary and outlook.

## Test Strategy

We have adopted the following strategy to evaluate available protocols and software products against the specific needs of backbone IP network providers:

1. Define a few sample applications for detailed accounting which are of actual interest to some members of the TF-TANT community.
2. Implement those applications with some of the available tools.
3. Evaluate the tools with respect to the suitability for the different tasks.
4. Provide feedback on possible improvements of the protocols and software products to the respective vendors/developers.

# Status of the Experiment

As of early September 1999, preliminary work has been performed as described in the following paragraphs.

The TF-TANT community has provided valuable input on how traffic accounting is currently used by DANTE and some NRNs, as well as on potential new applications.

DANTE has opened a test account on the test workstation in the Geneva TEN-155 PoP for use by the test participants.

The NetFlow accounting stream generated by the Geneva TEN-155 router has been diverted to a "flow replicator" program. This program, which has been developed by ???, is capable of copying a router's accounting stream and sending it to a set of receivers. This allows peaceful coexistence of the production statistics collector in the TEN-155 network and the programs evaluated within this experiment.

A selected set of software packages for the post-processing of NetFlow accounting data has been installed on the Geneva test workstation:

- CAIDA cflowd
- Cisco FlowCollector/FlowAnalyzer
- Fluxoscope (SWITCH's NetFlow accounting tool)

## Applications to Investigate

### Traffic Statistics at Exchange Points

One area where traditional traffic measurement is insufficient is where a network exchanges traffic with multiple peers over a shared medium such as a LAN-based Internet Exchange Point. Using per-interface counters, it is not possible to measure the traffic exchanged with individual peer networks.

Flow-based accounting can be used to overcome this by either

- using next/previous-hop IP or MAC address information
- using BGP routing information (neighbor/origin AS or AS paths)

BGP information can either be provided by the router itself (NetFlow v5 and later) or by the collecting process which interacts with a routing registry or a BGP speaker. Such possibilities should be studied in the parallel *Route Monitoring* experiment.

### Accounting for Volume-Based Charging

Network operators that use volume-based charging methods have found it problematic to charge an indiscriminate price for all traffic. Experience has shown that this leads to waste of resources, since people will save on local traffic which doesn't cost much to provide.

Therefore it is desirable to price traffic differentially according to destination/source. As an example, traffic within the NRN and with peers may be free (included in access fee), but traffic over expensive external

connections such as TEN-155 or a US transit provider might be charged for.

Such pricing schemes could be supported by flow-based accounting along these lines:

- Separate traffic counts are generated for each (customer,external-network) pair.
- Optionally, ``itemize" the counts for an organization according to internal cost centers (as done in JANET).
- Near-real-time feedback should be provided to organizations (or cost centers, or even individual users).
- A trail of individual accounting records is kept to substantiate bills. This trail can become quite large, so won't be around for very long. Therefore it is important to provide near-real-time feedback so that users can check their data frequently.

In addition, such a differentiated charging model may be extended to support differentiated services such as studied in the *DiffServ* experiment.

## Abuse/Attack Detection

Flow-based accounting can be used to detect anomalous traffic such as

- denial-of-service attacks such as ``smurf"
- attempts to breach security: systematic scans, known bugs/backdoors
- high amounts of non-adaptive traffic

Care must be taken to ensure privacy and to avoid false positives (for example, WWW caches may look similar to port scanners at first sight).

## Long-Term Traffic Analysis

As a prerequisite to intelligent connectivity and bandwidth provisioning, network operators want to be able to recognize large-scale trends about the traffic over their networks.

- application mix on different connections
- emerging applications
- interesting/important source/destination networks

## Detection of Routing Anomalies

Compare actual traffic flows with intended routing policy

The motivation behind this is that providers try to send their outbound traffic over the optimal links (to TEN-155 rather than the US, or towards settlement-free peers rather than paid transit connections), but they cannot easily exert control over the paths through inbound traffic comes back to their network.

## Intermediate Results

### Comparison Between Flow-Based Accounting Protocols

When comparing the IETF RTFM (Real-Time Flow Measurement) and Cisco NetFlow protocols, the following can be observed:

#### RTFM

RTFM is an IETF effort, which means that a priori it has a good potential to lead to a solution supported by multiple vendors. But so far, the only known implementations are Nevil Brownlee's NeTraMet system and another implementation from IBM.

The NeTraMet implementation runs on general-purpose computers under either MS-DOS or Unix. Data collection uses LAN interfaces such as Ethernet or FDDI in promiscuous mode. This makes it easy to deploy probes on shared LAN segments, but difficult to measure traffic over point-to-point links. As most of the interesting trunks in modern backbone networks run over such point-to-point links, this is a severe limitation. In particular we haven't found a way to test NeTraMet in a realistic setting within the TEN-155 network.

In the RTFM architecture, a measurement probe (traffic meter) is configured with a set of rules for classifying packets into flows. The Simple Network Management Protocol (SNMP) can be used both for installing those rulesets and for reading out measured data.

## Cisco NetFlow

NetFlow is proprietary to a router vendor (Cisco), but well-documented and has seen wide deployment in the short period since its first release. There is a relatively large (and growing) amount of software that can process NetFlow accounting information.

In NetFlow, data collection is performed at the router. If NetFlow is enabled on an interface, the packets *received* on that interface will be classified into flows according to a fixed set of parameters (source and destination IP address, protocol, source and destination TCP/UDP port, and TOS byte/DSCP). For each flow, octet and packet counters as well as some additional information is kept. Collected data for flows is then "exported" asynchronously to a specified management station. Whenever accounting data is exported for a flow, that flow is deleted from the flow cache. A flow can be deleted--and thus exported--for one of the following reasons:

1. The flow cache is full and a new flow must be created. In this case some kind of LRU scheme is probably used to determine which of the existing flows should be purged.
2. A packet has been received for the flow which signals the end of the flow. An example of this is a packet containing a TCP segment with the FIN bit set.
3. The lifetime of the flow in the cache has exceeded a limit. This limit used to be fixed at thirty minutes in early releases of the NetFlow router code. This is still the default in recent versions, but can be changed in a range of 1-60 minutes.
4. No new packets have been received for the flow for a certain amount of time. This *inactive timeout* can be configured to a range between 10 and 600 seconds in recent releases.

When flow accounting data is ready for export, the flow sender tries to "batch" multiple accounting records into a single UDP packet.

With the NetFlow v5 accounting format, every accounting record is 48 bytes long. Up to 30 flow accounting records are batched into a single UDP packet. A header of 24 bytes contains information about the entire set of flow accounting records in the packet.

The packet header includes a sequence number, so that missing packets can be detected. However, there is no possibility to have lost packets retransmitted.

Recent versions of Cisco IOS implement NetFlow accounting in "distributed" mode. In this mode, Versatile Interface Processors (VIPs) autonomously manage their own NetFlow caches, and export accounting data independently.

## Planned Work

The following steps are planned for the remaining lifetime of the experiment:

The different products should be configured and evaluated for the applications at hand.

More software packages might be investigated. In particular, the current set doesn't include any package which is specifically oriented towards charging and billing. However, those systems have recently started to become available and might be quite suitable for some of the applications, especially the charging application described in section [4.2](#).

Deployment issues should be investigated and documented. The resulting document should explain the tradeoffs in deciding where measurement probes should be deployed in the network, what must be measured, and so on.

The final report should include:

- the description of flow-based approaches to traffic accounting and analysis (RTFM, Cisco NetFlow)
- use of flow-based accounting for different applications (detailed description of deployment/scaling issues)
- comparison of tools and suggestions for improvement.

## Conclusions and Outlook

Flow-based accounting mechanisms such as RTFM or Cisco NetFlow have the potential to be very useful in short-term and long-term monitoring of backbone IP networks. Most of the applications that have been investigated so far are somewhat limited in the range of traffic categorization they support, which results in restricted usefulness for some of the applications envisioned. Some level of programmability would seem to be useful in the accounting data collection components. While it is possible to implement programmable data collection systems on general-purpose workstations, this may no longer be an option if categorization of accounting data is performed in the router, which may become necessary in some situations for performance reasons.

## References

<http://www.switch.ch/tf-tant/floma/> contains:

- proposal
- experiment status
- application scenarios
- pointers to software packages, related projects and other information

*Simon Leinen*

*Fri Sep 10 18:50:57 MET DST 1999*